

# Distributed Estimation of Dynamic Parameters : Regret Analysis

Shahin Shahrampour, Alexander Rakhlin and Ali Jadbabaie

**Abstract**—This paper addresses the estimation of a time-varying parameter in a network. A group of agents sequentially receive noisy signals about the parameter (or moving target), which does not follow any particular dynamics. The parameter is not observable to an individual agent, but it is globally identifiable for the whole network. Viewing the problem with an online optimization lens, we aim to provide the finite-time or non-asymptotic analysis of the problem. To this end, we use a notion of dynamic regret which suits the online, non-stationary nature of the problem. In our setting, dynamic regret can be recognized as a finite-time counterpart of stability in the mean-square sense. We develop a distributed, online algorithm for tracking the moving target. Defining the path-length as the consecutive differences between target locations, we express an upper bound on regret in terms of the path-length of the target and network errors. We further show the consistency of the result with static setting and noiseless observations.

## I. INTRODUCTION

Distributed detection, learning and estimation has been a main topic of interest in the past three decades [1]–[4]. With wide-spread applications in sensor, robotic, economic and social networks, distributed algorithms have received a considerable attention in science and engineering [5], [6]. In these scenarios, a group of agents aim to learn or estimate the value of a parameter. Each individual agent receives partially informative data about the parameter; however, the global spread of information in the network allows agents to accomplish the task collaboratively. Many of these information aggregation methods use *consensus* protocols as a crucial component [7], [8]. Distributed algorithms are popular for their ability to handle large data sets, low computation burden on agents and robustness to node failures.

On the other hand, online learning and optimization has been intensively studied in the literature of machine learning [9], [10], proving to be a powerful tool to model sequential decision problems. The problem can be modeled as a game between a learner and an adversary. The learner sequentially selects actions, and the adversary reveals the corresponding losses to the learner. The term *online* refers to the fact that the learner receives data in a sequential fashion. The popular performance metric for online algorithms is called *regret*.

This work was supported by ONR BRC Program on Decentralized, Online Optimization.

Shahin Shahrampour is with the Department of Electrical Engineering at Harvard University, Cambridge, MA 02138 USA. (e-mail: shahin@seas.harvard.edu).

Alexander Rakhlin is with the Department of Statistics at the University of Pennsylvania, Philadelphia, PA 19104 USA. (e-mail: rakhlin@wharton.upenn.edu).

Ali Jadbabaie is with the Department of Electrical and Systems Engineering at the University of Pennsylvania, Philadelphia, PA 19104 USA. (email: jadbabai@seas.upenn.edu).

Regret measures the performance of algorithm versus a pre-defined benchmark. For instance, the benchmark could be the best fixed action had the learner known all the losses in advance. In a broad sense, when the benchmark is a fixed sequence, the regret is called *static*, whereas a *time-varying* benchmark sequence brings forward the notion of *dynamic* regret [10]–[12].

The goal of this paper is to develop a *distributed, online* algorithm for tracking *time-varying* parameters. To this end, a unifying observation is to view the distributed estimation problem as an online optimization. A network of agents aim to track a moving target which is only partially observable to each agent. The parameter can represent a sequence of high-dimensional data where agents only receive a low-dimensional version of it. Therefore, they must communicate with each other to track the target. Each time agents form their estimates of the parameter, the whole network incurs a loss (network loss). We formalize the problem as an instance of distributed, online optimization whose *dynamic* regret can be characterized in the general form of

$$\text{Regret} = \text{Decentralized Loss} - \text{Centralized Loss}.$$

In other words, the regret compares the performance of the decentralized algorithm versus its centralized counterpart.

Our main contribution is to provide the *finite-time* or *non-asymptotic* analysis of the problem using the notion of regret. In this context, regret can also be interpreted as a finite-time analog of *stability* in mean-square sense. In contrast to existing approaches, we do not restrict the dynamics of the parameter, i.e., the target is allowed to move along an arbitrary trajectory. The freedom given to the dynamics broadens the application of the problem. For instance, the parameter can represent the price of a product which does not have a particular pattern, or the location of a vehicle that moves along an unpredictable trajectory. In turn, our estimation update is akin to *consensus+innovation* updates in the literature with an important distinction that it does not assume *any* dynamics on the parameter.

We then characterize the behavior of regret in finite-time regime. To this end, we quantify the *path-length* in terms of the consecutive differences between target locations, and prove that the regret can be bounded in terms of the path-length of the target. Interestingly, our regret bound interpolates smoothly from static to dynamic setting. That is, when the parameter is static, we can recover the corresponding results with *static regret* (with appropriate tuning of problem parameters). We further simplify the bound in two cases: static target and noiseless observation.

## Related Work

There exists a large body of literature on estimation of dynamic parameters. These works mostly assume that the parameter follows a *known* dynamics, say, the model can be represented as an LTI system. Perhaps the most classical example is the celebrated Kalman filtering [13]. The elegance of Kalman filter has motivated a lot of researchers to investigate the problem in the context of networks. We cannot hope to do the justice to the extensive literature on *distributed* Kalman filtering, and refer the avid reader to a series of works on this topic [14]–[17]. The works of [18]–[21] are also in the same spirit in which the parameter follows linear dynamics. Acemoglu et al. [18] present a rule-of-thumb learning rule, and provide the *asymptotic* behavior of their update. In [19], Khan et al. investigate the trade-off between *stability* of the linear dynamics and *connectivity* of the network. They show that their update can potentially track unstable linear models driven by noise. Similarly, an algorithm consisting of pseudo-innovations is developed in [20]. Moreover, the authors of [21] consider a *scalar* linear model, and characterize an explicit expression for the mean-square deviation. Restricting their attention to noiseless case, De et al. pose an *inverse* problem in [22]. In their model, the parameter can be observed whereas the dynamics is *unknown* and must be estimated. On the other hand, Atanasov et al. [23] propose a distributed, linear estimator in wireless sensor networks which encompasses an auxiliary localization procedure. The authors prove the mean-square *consistency* of the joint localization and target estimation algorithm. Moreover, Sayin et al. [24] present a distributed, online algorithm for state estimation applicable to Big Data. The authors of [25] develop an optimization-based, prediction-correction method for tracking the trajectory of a moving target. Finally, our work is also related to [26], where the problem is considered in the context of social networks.

## Organization

The rest of this paper is organized as follows. In the next section we introduce the notation, and formalize the estimation problem using the notion of regret. In Section III, we present our technical results and their consequences. Notably, we demonstrate a path-length bound on regret. Section IV consists of concluding remarks, and in Section V we provide the proof of our technical results.

## II. PROBLEM FORMULATION

### A. Nomenclature

We adhere to the following notation in the exposition of our results:

$[n]$	The set $\{1, 2, \dots, n\}$ for any integer $n$
$x^\top$	Transpose of the vector $x$
$I_d$	Identity matrix of size $d \times d$
$\ \cdot\ $	The spectral norm operator
$\mathbb{1}_d$	Vector of all ones with dimension $d$
$\mathbf{0}_d$	Vector of all zeros with dimension $d$
$\lambda_i(P)$	The $i$ -th largest eigenvalue of matrix $P$
$\rho(P)$	The spectral radius of matrix $P$

Throughout, the variables in bold are random, and all the vectors are in column form. When referring to vector of all ones or zeros, sometimes the dimension is omitted and can be inferred from the context.

### B. Observation Model and Regret Definition

Our objective is to track a *time-varying* parameter that is not confined to a certain dynamics. The parameter could represent the location of a moving target following an arbitrary trajectory. We denote the parameter by  $\theta_t \in \mathbb{R}^d$  at each time  $t \in [T]$ . A network of  $n$  agents collaborate with each other to track the parameter which is only *partially* observable to each agent. More formally, the observation model for agent  $i \in [n]$  can be expressed in the following form

$$\mathbf{y}_{i,t} = H_i \theta_t + \mathbf{w}_{i,t}, \quad (1)$$

where  $H_i \in \mathbb{R}^{m_i \times d}$  and  $\mathbf{w}_{i,t} \in \mathbb{R}^{m_i}$  denote the observation matrix and noise, respectively. For instance,  $\theta_t$  can be a high-dimensional data where agents only observe a low-dimensional version of that, i.e.  $m_i$  might be much smaller than  $d$ . We respect the standard assumption of zero-mean and finite-covariance for the observation noise. That is, we assume that  $\mathbb{E}[\mathbf{w}_{i,t}] = \mathbf{0}$  and  $\mathbb{E}[\mathbf{w}_{i,t}^\top \mathbf{w}_{i,t}] = W_i$ , respectively. We further assume the independence over time and space; that is,  $\mathbb{E}[\mathbf{w}_{i,t}^\top \mathbf{w}_{j,s}] = 0$  for  $i \neq j$  or  $s \neq t$ .

Note that when the parameter is *static*, i.e. when  $\theta_t = \theta$ , we recover the classical distributed estimation problem (see e.g. [2]). Though not locally observable to each agent, we assume that the parameter is *globally* identifiable from the standpoint of all agents in the network.

*Assumption 1:* The sequence  $\{\theta_t\}_{t=1}^T$  is globally identifiable, i.e., it holds that the matrix

$$H = \frac{1}{n} \sum_{i=1}^n H_i^\top H_i,$$

is invertible.

The estimation procedure follows an online protocol. At time  $t$ , each agent  $i$  forms an estimate  $\hat{\theta}_{i,t}$  of  $\theta_t$  based on observations  $\{\mathbf{y}_{i,\tau}\}_{\tau=1}^{t-1}$ . After that, the new signal  $\mathbf{y}_{i,t}$  becomes available to the agent. The online nature of our estimation problem allows us to pose it as an instance of online optimization. Therefore, before deriving the explicit update for  $\hat{\theta}_{i,t}$ , we need to introduce a few notions based on the literature of online optimization. Let us define a *local* square loss

$$\ell_{i,t}(\theta) := \mathbb{E} \left[ \|\mathbf{y}_{i,t} - H_i \theta\|^2 \right], \quad (2)$$

for each agent  $i \in [n]$ , followed by the *network* loss as

$$\ell_t(\theta) := \sum_{i=1}^n \mathbb{E} \left[ \|\mathbf{y}_{i,t} - H_i \theta\|^2 \right]. \quad (3)$$

Note that Assumption 1 guarantees that  $\ell_t(\theta)$  has a unique minimizer at each time  $t$ . Agents aim to minimize the

accumulated network loss over time. Equivalently, the goal of the network is to minimize the *regret* defined as

$$\text{Reg}_T := \frac{1}{T} \sum_{t=1}^T \left( \frac{1}{n} \sum_{j=1}^n \ell_t(\hat{\theta}_{j,t}) - \ell_t(\theta_t) \right). \quad (4)$$

The regret quantifies the difference between the *realized* and *ideal* network loss: the estimation  $\hat{\theta}_{j,t}$  of each agent  $j$  is evaluated at the function  $\ell_t(\cdot)$ , and the average (among agents) is subtracted by the loss of exact, ideal estimate  $\theta_t$ . Evidently, the ideal estimate (solution to the network loss (3)) is not available to an individual agent; hence, one can also interpret the regret as the difference between an algorithm that receives local loss (decentralized) versus another algorithm that accesses the network loss (centralized).

### C. Network Model and Estimation Update

The interactions of agents, which in turn defines the network, is captured with the matrix  $P$ . Formally, we denote by  $[P]_{ij}$ , the  $ij$ -th entry of the matrix  $P$ . When  $[P]_{ij} > 0$ , agent  $i$  communicates with agent  $j$ . We assume that  $P$  is symmetric, stochastic with positive diagonal elements. The assumption simply guarantees the information flow in the network. Alternatively, from technical point of view, we respect the following hypothesis.

*Assumption 2:* We assume the Markov chain  $P$  is irreducible and aperiodic, and  $P^\top = P$ .

The assumption implies that the Markov chain  $P$  has a unique stationary distribution i.e.,  $\mathbb{1}^\top P = \mathbb{1}^\top$  is the unique (unnormalized) left eigenvector corresponding to  $\lambda_1(P) = 1$ . It also entails that  $\lambda_1(P)$  is unique, and the other eigenvalues of  $P$  are less than unit in magnitude [27].

To follow the trajectory of the parameter, we propose an update rule which can be cast as a *distributed* variant of Online Gradient Descent [10] with *noisy* feedback. It takes the form

$$\hat{\theta}_{i,t} = \sum_{j=1}^n [P]_{ij} \hat{\theta}_{j,t-1} + \alpha H_i^\top \left( \mathbf{y}_{i,t-1} - H_i \hat{\theta}_{i,t-1} \right), \quad (5)$$

where  $\alpha \in \mathbb{R}$  is the step size. The update is akin to *consensus+innovation* updates in the literature (see e.g. [19], [28]) with a distinction that it does *not* assume any dynamics on the parameter  $\theta_t$ . The consensus part forces agents to keep their estimates close to each other, and the innovation part takes into account the new observation. We can now shed light on our previous interpretation of regret: had the agents known the network loss  $\ell_t(\theta)$ , they could have formed estimates based on Online Gradient Descent (which is a centralized algorithm). Since agents only know the local loss (2), regret (4) is comparing the distributed algorithm versus its centralized counterpart.

We further define the *path-length* of the moving target as

$$C_T := \sum_{t=1}^T \|\theta_t - \theta_{t-1}\|^2, \quad (6)$$

which indicates how much the parameter drifts over time. In particular, when bounding the regret (4), the path-length  $C_T$

becomes useful. As we shall explore in the next section, there are connections between mean-square stability and regret; hence, we close this section by defining the mean-square stability.

*Definition 1:* We say the process is stable in the mean-square sense, whenever

$$\frac{1}{n} \sum_{i=1}^n \mathbb{E} \left[ \|\hat{\theta}_{i,t} - \theta_t\|^2 \right] \rightarrow \sigma,$$

where  $\sigma \geq 0$  is a finite constant.

## III. MAIN RESULTS

In this section, we provide our technical results and their consequences. More specifically, our objective is to find an anytime bound on the regret involving the path-length of target. We further interpret the bound in several circumstances.

### A. Error Process and Stability

Stability (in mean or mean-square sense) is closely attached to the behavior of estimation error. Let us define the *local* error process as follows,

$$\mathbf{e}_{i,t} := \hat{\theta}_{i,t} - \theta_t. \quad (7)$$

Stacking the local errors in a vector, we represent the *global* error by

$$\mathbf{e}_t := [\mathbf{e}_{1,t}, \dots, \mathbf{e}_{n,t}]^\top. \quad (8)$$

In the following lemma, we show that the error process can be represented as an LTI system.

*Lemma 1:* The error process (8) can be characterized via an LTI system as follows

$$\mathbf{e}_t = Q \mathbf{e}_{t-1} + \mathbf{u}_t,$$

where

$$Q := P \otimes I_d - \alpha \text{diag} [H_1^\top H_1, \dots, H_n^\top H_n],$$

and

$$\mathbf{u}_t := \mathbb{1}_n \otimes (\theta_{t-1} - \theta_t) + \alpha \begin{bmatrix} H_1^\top \mathbf{w}_{1,t-1} \\ \vdots \\ H_n^\top \mathbf{w}_{n,t-1} \end{bmatrix}.$$

We observe from Lemma 1 that to ensure mean stability (to have  $\mathbb{E}[\mathbf{e}_t] \rightarrow \mathbf{0}$  as  $t \rightarrow \infty$ ), the conditions

$$\rho(Q) < 1 \quad \text{and} \quad \mathbb{E}[\theta_t - \theta_{t-1}] \rightarrow \mathbf{0},$$

need to be satisfied. To push the spectral radius of  $Q$  inside the unit circle, we can simply tune the step size  $\alpha$ . We quantify this condition in Section III-C, and show the role of global identifiability (Assumption 1) and connectivity of network (Assumption 2) in existence of such  $\alpha$ .

However, the second condition entirely depends on the problem environment. For instance, the condition is evidently satisfied in the static case ( $\theta_t = \theta$ ). It also holds when  $\theta_t$  is generated randomly based on a *stationary* distribution, i.e. whenever  $\mathbb{E}[\theta_t] = \theta$ . Regardless of stability, we can always characterize the regret behavior in terms of path-length of the target.

### B. A Path-Length Bound for Regret

In this section, we analyze the regret (4), which is a measure of *finite-time* performance. The following lemma exhibits the connection of mean-square stability (Definition 1) and regret (4). In particular, the lemma is used to bound the regret in terms of path-length of the target (6).

*Lemma 2:* The regret defined in (4) can be bounded as,

$$\text{Reg}_T \leq \frac{1}{nT} \sum_{t=1}^T \sum_{i=1}^n \|H_i\|^2 \mathbb{E} [\|\mathbf{e}_t\|^2].$$

Lemma 2 proves that stability implies boundedness of asymptotic regret since

$$\mathbb{E} [\|\mathbf{e}_t\|^2] = \sum_{i=1}^n \mathbb{E} [\|\hat{\theta}_{i,t} - \theta_t\|^2],$$

and once the limit exists, the Cesàro mean preserves it. However, we are interested in non-asymptotic analysis of the problem. We present our finite-time statement in the subsequent theorem.

*Theorem 1:* Let the sequence  $\{\theta_t\}_{t=1}^T$  be globally identifiable (Assumption 1), and the Markov chain  $P$  be irreducible and aperiodic (Assumption 2). If each agent  $i \in [n]$  generates the estimate sequence  $\{\hat{\theta}_{i,t}\}_{t=1}^T$  according to the update rule (5), the regret satisfies

$$\begin{aligned} \text{Reg}_T \leq & \frac{1}{n} \sum_{i=1}^n \|H_i\|^2 \frac{\alpha^2 \sum_{i=1}^n \|H_i\|^2 W_i}{1 - \|Q\|} \\ & + \frac{1}{T} \sum_{i=1}^n \|H_i\|^2 \frac{C_T}{(1 - \|Q\|)^2}. \end{aligned}$$

Theorem 1 provides a path-length bound for regret. The underlying intuition behind the term  $C_T$  is as follows: as agents decide on the next value based on previous observations (online prediction), they are always one step behind in estimation. Even when the observations are *noiseless* ( $W_i = 0$  for all  $i \in [n]$ ), the second term still remains, which is an artifact of agent  $i$  using  $\mathbf{y}_{i,t-1}$  to predict  $\theta_t$ .

We remark that path-length regret bounds were previously studied in the context of online optimization for *centralized* frameworks (see e.g. [10]–[12]). Here, we specialized to quadratic losses, and proved a path-length bound in *distributed* setting. Therefore, our bound involves an additional network penalty comparing to the centralized framework.

### C. Tuning the Step Size

We now discuss several aspects of the regret bound derived in Theorem 1. To this end, we need to clarify the dependence of  $\|Q\|$  on step size  $\alpha$  to analyze the bound. Recall the closed-form of  $Q$  in Lemma 1 as well as stochasticity of  $P$  which implies  $P\mathbb{1}_n = \mathbb{1}_n$ ; pre/post multiply  $Q$  by a vector of all ones with dimension  $nd$  ( $\mathbb{1} = \mathbb{1}_n \otimes \mathbb{1}_d$ ) to get

$$\begin{aligned} \mathbb{1}^\top Q \mathbb{1} &= \mathbb{1}^\top (P \otimes I_d - \alpha \text{diag} [H_1^\top H_1, \dots, H_n^\top H_n]) \mathbb{1} \\ &= \mathbb{1}^\top (\mathbb{1} - \alpha \text{diag} [H_1^\top H_1, \dots, H_n^\top H_n] \mathbb{1}) \\ &= \mathbb{1}^\top \mathbb{1} - \alpha \mathbb{1}^\top \text{diag} [H_1^\top H_1, \dots, H_n^\top H_n] \mathbb{1} \\ &= \mathbb{1}^\top \mathbb{1} - n\alpha \mathbb{1}_d^\top H \mathbb{1}_d, \end{aligned}$$

as  $H = \frac{1}{n} \sum_{i=1}^n H_i^\top H_i$ . Therefore, since  $\mathbb{1}^\top \mathbb{1} = nd$ , we obtain

$$\frac{\mathbb{1}^\top Q \mathbb{1}}{\mathbb{1}^\top \mathbb{1}} = 1 - n\alpha \frac{\mathbb{1}_d^\top H \mathbb{1}_d}{\mathbb{1}^\top \mathbb{1}} = 1 - \alpha \frac{\mathbb{1}_d^\top H \mathbb{1}_d}{\mathbb{1}_d^\top \mathbb{1}_d}.$$

In view of Assumption 2,  $\lambda_1(P) = 1$  is unique, and no direction other than  $\mathbb{1}_n$  can recover the trivial eigenvalue. Hence, depending on the null space of  $\text{diag} [H_1^\top H_1, \dots, H_n^\top H_n]$ ,  $\alpha$  can be set small enough such that the following upper bound holds

$$\lambda_1(Q) \leq 1 - \alpha \lambda_n(H). \quad (9)$$

By the same token, Weyl's eigenvalue inequality implies a lower bound the smallest eigenvalue of  $Q$  as

$$\lambda_n(Q) \geq \lambda_n(P) - \alpha \lambda_1(\text{diag} [H_1^\top H_1, \dots, H_n^\top H_n]). \quad (10)$$

Once again choosing small enough  $\alpha$  guarantees that the RHS of (9) is larger than the absolute value of the RHS of (10). Let us distinguish such regime as  $\alpha \leq \alpha_{\max}$ . Then, symmetry of  $Q$  warrants that

$$\|Q\| \leq 1 - \alpha \lambda_n(H), \quad \forall \alpha \leq \alpha_{\max}.$$

Therefore, disregarding the constants (dependence on  $\{H_i\}_{i=1}^n$  and  $H$ ) in Theorem 1, we can simplify the regret bound using above as follows

$$\text{Reg}_T \leq \mathcal{O} \left( \alpha \sum_{i=1}^n W_i + \frac{C_T}{T\alpha^2} \right).$$

The following comments are now in order:

- If the target is fixed, i.e.  $\theta_t = \theta$  for every  $t \in [T]$ , we have  $C_T = 0$ , and the second term in the bound vanishes. Then, we can set  $\alpha = \min\{\frac{1}{T}, \alpha_{\max}\}$  to maintain the  $\mathcal{O}(\frac{1}{T})$  rate. This choice of step size and the corresponding result is consistent with the results in *static* setting.
- When the observations are noiseless ( $W_i = 0$ ) the first term becomes zero, and  $\alpha = \min\{1, \alpha_{\max}\}$  recovers the  $\mathcal{O}(\frac{C_T}{T})$  rate. In this regime, a sub-linear path-length  $C_T$  always guarantees a zero asymptotic regret.
- In the general case tuning  $\alpha = \min\{C_T^{1/3} T^{-1/3}, \alpha_{\max}\}$  yields a regret of  $\mathcal{O}(C_T^{1/3} T^{-1/3})$  which holds for any  $T$ .

### IV. CONCLUDING REMARKS

In this paper, we considered the distributed estimation problem in dynamic environments. A network of agents partially observe the parameter of interest which is time-varying with no particular dynamics. However, the parameter is globally observable from the standpoint of all agents together. We pose the problem as an instance of distributed, online optimization, where agents suffer some loss after each round of estimation. In turn, they aim to minimize the network loss defined as the sum of individual losses. We formulated the problem using the notion of dynamic regret from online optimization literature. Our main contribution was to provide the non-asymptotic analysis of the dynamic



regret. We first showed that the regret can be related to the finite-time counterpart of stability. Next, we defined the path-length as the consecutive differences between target locations, and derived an upper bound on regret in terms of the path-length. We finally demonstrated that our bound is consistent with special cases of static setting and noiseless observations.

There are several directions in which our work can be improved. Our setting does not include any prior assumption on the dynamics of the parameter. While the generality of this setup is appealing, it is also interesting to investigate the problem for particular dynamics. More specifically, once agents know the dynamics of the parameter (even in the non-linear case) they can incorporate it into the algorithm, and obtain more accurate estimates. Furthermore, the choice of step size in the general case requires a prior knowledge of the path-length. An alternative approach is to have an online feedback of the trajectory as the algorithm moves forward. Therefore, an adaptive, online algorithm for tracking dynamic parameters is still an open problem.

## V. APPENDIX : PROOFS

**Proof of Lemma 1.** In view of equations (1) and (5), we write

$$\begin{aligned}\hat{\theta}_{i,t} - \theta_t &= \sum_{j=1}^n [P]_{ij} \hat{\theta}_{j,t-1} - \theta_t \\ &\quad + \alpha H_i^\top \left( \mathbf{y}_{i,t-1} - H_i \hat{\theta}_{i,t-1} \right) \\ &= \sum_{j=1}^n [P]_{ij} (\hat{\theta}_{j,t-1} - \theta_{t-1}) + \theta_{t-1} - \theta_t \\ &\quad + \alpha H_i^\top \left( H_i \theta_{t-1} + \mathbf{w}_{i,t-1} - H_i \hat{\theta}_{i,t-1} \right), \quad (11)\end{aligned}$$

since  $\sum_{j=1}^n [P]_{ij} = 1$ . Rewriting above based on the definition of local error process (7), we obtain

$$\begin{aligned}\mathbf{e}_{i,t} &= \sum_{j=1}^n [P]_{ij} \mathbf{e}_{j,t-1} - \alpha H_i^\top H_i \mathbf{e}_{i,t-1} \\ &\quad + \theta_{t-1} - \theta_t + \alpha H_i^\top \mathbf{w}_{i,t-1}.\end{aligned}$$

Representing above in the matrix form using the Kronecker product completes the proof. ■

**Proof of Lemma 2.** Note equations (1) and (3) to observe

$$\ell_t(\theta_t) = \sum_{i=1}^n \mathbb{E} [\|\mathbf{w}_{i,t}\|^2] = \sum_{i=1}^n W_i, \quad (12)$$

and

$$\begin{aligned}\ell_t(\hat{\theta}_{j,t}) &= \sum_{i=1}^n \mathbb{E} \left[ \|\mathbf{y}_{i,t} - H_i \hat{\theta}_{j,t}\|^2 \right] \\ &= \sum_{i=1}^n \mathbb{E} \left[ \|H_i \theta_t + \mathbf{w}_{i,t} - H_i \hat{\theta}_{j,t}\|^2 \right] \\ &= \sum_{i=1}^n \mathbb{E} \left[ \|H_i \mathbf{e}_{j,t} - \mathbf{w}_{i,t}\|^2 \right],\end{aligned}$$

where we used (7) in the last step. Given that noise realizations are independent over time, we can simplify above

$$\begin{aligned}\ell_t(\hat{\theta}_{j,t}) &= \sum_{i=1}^n \mathbb{E} \left[ \|H_i \mathbf{e}_{j,t} - \mathbf{w}_{i,t}\|^2 \right] \\ &= \sum_{i=1}^n \left( \mathbb{E} [\|H_i \mathbf{e}_{j,t}\|^2] + W_i \right).\end{aligned} \quad (13)$$

Therefore, combining (12) and (13), we get

$$\begin{aligned}\frac{1}{n} \sum_{j=1}^n \ell_t(\hat{\theta}_{j,t}) - \ell_t(\theta_t) &= \frac{1}{n} \sum_{j=1}^n \sum_{i=1}^n \mathbb{E} [\|H_i \mathbf{e}_{j,t}\|^2] \\ &\leq \frac{1}{n} \sum_{j=1}^n \sum_{i=1}^n \|H_i\|^2 \mathbb{E} [\|\mathbf{e}_{j,t}\|^2] \\ &= \left( \frac{1}{n} \sum_{i=1}^n \|H_i\|^2 \right) \mathbb{E} [\|\mathbf{e}_t\|^2],\end{aligned}$$

where we used (8) in the last line. Summing over  $t \in [T]$  completes the proof. ■

**Proof of Theorem 1.** In view of Lemma 1, we have

$$\begin{aligned}\mathbb{E} [\|\mathbf{e}_t\|^2] &\leq \|Q\|^2 \mathbb{E} [\|\mathbf{e}_{t-1}\|^2] \\ &\quad + \alpha^2 \sum_{i=1}^n \|H_i\|^2 W_i + n \|\theta_{t-1} - \theta_t\|^2 \\ &\quad + 2[\mathbb{1}_n \otimes (\theta_{t-1} - \theta_t)]^\top Q \mathbb{E}[\mathbf{e}_{t-1}]. \quad (14)\end{aligned}$$

since  $\mathbb{E}[\mathbf{w}_{i,t-1}] = \mathbf{0}$ . Using the simple fact that for any  $\beta > 0$

$$2ab \leq \frac{1}{\beta} a^2 + \beta b^2,$$

based on AM-GM inequality, we apply the Cauchy-Schwarz inequality to bound

$$\begin{aligned}2[\mathbb{1}_n \otimes (\theta_{t-1} - \theta_t)]^\top Q \mathbb{E}[\mathbf{e}_{t-1}] &\leq \\ \frac{1}{\beta} \|\mathbb{1}_n \otimes (\theta_{t-1} - \theta_t)\|^2 + \beta \|Q\|^2 \mathbb{E}[\|\mathbf{e}_{t-1}\|^2],\end{aligned}$$

for any  $\beta > 0$ , and simplify (14) to

$$\begin{aligned}\mathbb{E} [\|\mathbf{e}_t\|^2] &\leq (1 + \beta) \|Q\|^2 \mathbb{E} [\|\mathbf{e}_{t-1}\|^2] \\ &\quad + \alpha^2 \sum_{i=1}^n \|H_i\|^2 W_i + \frac{\beta + 1}{\beta} n \|\theta_{t-1} - \theta_t\|^2,\end{aligned}$$

which entails

$$\begin{aligned}&\left( 1 - (1 + \beta) \|Q\|^2 \right) \mathbb{E} [\|\mathbf{e}_t\|^2] \\ &\leq (1 + \beta) \|Q\|^2 \left( \mathbb{E} [\|\mathbf{e}_{t-1}\|^2] - \mathbb{E} [\|\mathbf{e}_t\|^2] \right) \\ &\quad + \alpha^2 \sum_{i=1}^n \|H_i\|^2 W_i + \frac{\beta + 1}{\beta} n \|\theta_{t-1} - \theta_t\|^2. \quad (15)\end{aligned}$$

Using the convention  $\mathbb{E} [\|\mathbf{e}_0\|^2] = 0$ , the following sum telescopes

$$\sum_{t=1}^T \mathbb{E} [\|\mathbf{e}_{t-1}\|^2] - \mathbb{E} [\|\mathbf{e}_t\|^2] \leq \mathbb{E} [\|\mathbf{e}_0\|^2] = 0,$$

and summing (15) over  $t \in [T]$ , we derive

$$\sum_{t=1}^T \mathbb{E} [\|\mathbf{e}_t\|^2] \leq \frac{\alpha^2 T \sum_{i=1}^n \|H_i\|^2 W_i}{1 - (1 + \beta)\|Q\|^2} + \frac{\beta + 1}{\beta} \frac{n \sum_{t=1}^T \|\theta_{t-1} - \theta_t\|^2}{1 - (1 + \beta)\|Q\|^2}.$$

Recall that  $0 < \beta < \|Q\|^{-2} - 1$  is a free parameter in the analysis. Letting  $\beta = \|Q\|^{-1} - 1$ , and recalling Definition 6, we calculate the bound above to get

$$\sum_{t=1}^T \mathbb{E} [\|\mathbf{e}_t\|^2] \leq \frac{\alpha^2 T \sum_{i=1}^n \|H_i\|^2 W_i}{1 - \|Q\|} + \frac{n C_T}{(1 - \|Q\|)^2}.$$

Appealing to Lemma 2 completes the proof. ■

## REFERENCES

- [1] J. N. Tsitsiklis, "Decentralized detection by a large number of sensors," *Mathematics of Control, Signals and Systems*, vol. 1, no. 2, pp. 167–182, 1988.
- [2] S. S. Stanković, M. S. Stanković, and D. M. Stipanović, "Decentralized parameter estimation by consensus based stochastic approximation," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 531–543, 2011.
- [3] K. Drakopoulos, A. Ozdaglar, and J. N. Tsitsiklis, "On learning with finite memory," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6859–6872, 2013.
- [4] S. Shahrampour, A. Rakhlin, and A. Jadbabaie, "Distributed detection: Finite-time analysis and impact of network topology," *IEEE Transactions on Automatic Control*, vol. 61, 2016.
- [5] F. Bullo, J. Cortés, and S. Martinez, *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms: A Mathematical Approach to Motion Coordination Algorithms*. Princeton University Press, 2009.
- [6] D. Acemoglu, M. A. Dahleh, I. Lobel, and A. Ozdaglar, "Bayesian learning in social networks," *The Review of Economic Studies*, vol. 78, no. 4, pp. 1201–1236, 2011.
- [7] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.
- [8] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.
- [9] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge University Press, 2006.
- [10] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *International Conference on Machine Learning*, 2003.
- [11] E. Hall and R. Willett, "Online convex optimization in dynamic environments," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 647–662, June 2015.
- [12] A. Jadbabaie, A. Rakhlin, S. Shahrampour, and K. Sridharan, "Online optimization: Competing with dynamic comparators," in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, 2015, pp. 398–406.
- [13] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [14] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filters," in *IEEE Conference on Decision and Control (CDC), European Control Conference (ECC)*, 2005, pp. 8179–8184.
- [15] —, "Distributed kalman filtering for sensor networks," in *IEEE Conference on Decision and Control (CDC)*, 2007, pp. 5492–5498.
- [16] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Distributed kalman filtering based on consensus strategies," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 622–633, 2008.
- [17] R. Olfati-Saber, "Kalman-consensus filter: Optimality, stability, and performance," in *IEEE Conference on Decision and Control (CDC) held jointly with Chinese Control Conference (CCC)*, 2009, pp. 7036–7042.
- [18] D. Acemoglu, A. Nedić, and A. Ozdaglar, "Convergence of rule-of-thumb learning rules in social networks," in *IEEE Conference on Decision and Control (CDC)*, 2008, pp. 1714–1720.
- [19] U. Khan, S. Kar, A. Jadbabaie, J. M. Moura *et al.*, "On connectivity, observability, and stability in distributed estimation," in *IEEE Conference on Decision and Control (CDC)*, 2010, pp. 6639–6644.
- [20] S. Das and J. M. Moura, "Distributed state estimation in multi-agent networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 4246–4250.
- [21] S. Shahrampour, S. Rakhlin, and A. Jadbabaie, "Online learning of dynamic parameters in social networks," in *Advances in Neural Information Processing Systems*, 2013.
- [22] A. De, S. Bhattacharya, P. Bhattacharya, N. Ganguly, and S. Chakrabarti, "Learning a linear influence model from transient opinion dynamics," in *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*. ACM, 2014, pp. 401–410.
- [23] N. Atanasov, R. Tron, V. M. Preciado, and G. J. Pappas, "Joint estimation and localization in sensor networks," in *IEEE Conference on Decision and Control (CDC)*, 2014, pp. 6875–6882.
- [24] M. O. Sayin, N. D. Vanli, I. Delibalta, and S. S. Kozat, "Optimal and efficient distributed online learning for big data," in *IEEE International Congress on Big Data (BigData Congress)*, 2015, pp. 126–133.
- [25] A. Simonetto, A. Mokhtari, A. Koppel, G. Leus, and A. Ribeiro, "A decentralized prediction-correction method for networked time-varying convex optimization," in *Proceedings of the 6th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (Cancun, Mexico)*, 2015.
- [26] R. van Oosten, "Learning from neighbors in a changing world," 2016.
- [27] J. S. Rosenthal, "Convergence rates for markov chains," *SIAM Review*, vol. 37, no. 3, pp. 387–405, 1995.
- [28] S. Kar, J. M. Moura, and K. Ramanan, "Distributed parameter estimation in sensor networks: Nonlinear observation models and imperfect communication," *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3575–3605, 2012.